Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	103	munter.inv.	US-PGPUB; USPAT	OR	OFF	2005/08/10 15:28
L2	659	vickers.inv.	US-PGPUB; USPAT	OR	OFF	2005/08/10 15:28
L4	62	beshai.inv.	US-PGPUB; USPAT	OR	OFF	2005/08/10 15:28
L5	806	1 or 2 or 4	US-PGPUB; USPAT	OR	OFF	2005/08/10 15:29
L6	57430	(multiple near2 channel$1) multi$1channel$1	US-PGPUB; USPAT	OR	OFF	2005/08/10 15:29
L7	57	L6 and 5	US-PGPUB; USPAT	OR	OFF	2005/08/10 15:29
L8	29	delay.clm. and propagation.clm. and latenc$3.clm.	US-PGPUB	OR	OFF	2005/08/10 15:29
L9	0	L8 and 5	US-PGPUB; USPAT	OR	OFF	2005/08/10 15:29
L10	48	5 and nortel.as.	US-PGPUB; USPAT	OR	OFF	2005/08/10 15:29
L11	24	10 and link.clm.	US-PGPUB; USPAT	OR	OFF	2005/08/10 15:30
L12	31	10 and link$1.clm.	US-PGPUB; USPAT	OR	OFF	2005/08/10 15:30
S1	57060	(multiple near2 channel$1) multi$1channel$1	US-PGPUB; USPAT	OR	OFF	2005/08/09 13:59
S2	11845	load adj balanc$3	US-PGPUB; USPAT	OR	OFF	2005/07/27 14:22
S3	60	S1 same S2	US-PGPUB; USPAT	OR	OFF	2005/07/27 14:22
S4	172157	propagation delay and S3	US-PGPUB; USPAT	OR	OFF	2005/07/27 14:22
S5	32	(propagation delay) and S3	US-PGPUB; USPAT	OR	OFF	2005/07/27 14:26
S6	3405459	@ad<"20001201" @rlad<"20001201"	US-PGPUB; USPAT	OR	OFF	2005/07/27 14:26
S7	17	S5 and S6	US-PGPUB; USPAT	OR	OFF	2005/07/27 14:27
S8	727135	(delay$3 wait$3 time tim$3 period) same (vacanc$3 bandwidth$1 flow$3 data$1flow$3)	US-PGPUB; USPAT	OR	OFF	2005/08/09 14:00
S9	3116426	(add$3 sum summing summation combin$3 combination$1)	US-PGPUB; USPAT	OR	OFF	2005/08/09 14:01
S10	124721	S8 same S9	US-PGPUB; USPAT	OR	OFF	2005/08/09 14:01

S11	3407594	@ad<"20001201" @rlad<"20001201"	US-PGPUB; USPAT	OR	OFF	2005/08/09 14:01
S12	94186	S10 and S11	US-PGPUB; USPAT	OR	OFF	2005/08/09 14:01
S13	83	S10 same load adj balanc$4	US-PGPUB; USPAT	OR	OFF	2005/08/09 15:20
S20	1	dennison.XA. AND VAUGHN.XP.	US-PGPUB; USPAT	OR	OFF	2005/08/10 13:34
S21	699	718/105.ccls.	US-PGPUB; USPAT	OR	OFF	2005/08/09 15:57
S22	3116426	(add$3 sum summing summation combin$3 combination$1)	US-PGPUB; USPAT	OR	OFF	2005/08/09 15:57
S23	633	S21 and S22	US-PGPUB; USPAT	OR	OFF	2005/08/09 15:58
S24	727135	(delay$3 wait$3 time tim$3 period) same (vacanc$3 bandwidth$1 flow$3 data$1flow$3)	US-PGPUB; USPAT	OR	OFF	2005/08/09 15:58
S25	124721	S24 same S22	US-PGPUB; USPAT	OR	OFF	2005/08/09 15:58
S26	66	S21 and S25	US-PGPUB; USPAT	OR	OFF	2005/08/09 15:58
S27	3407594	@ad<"20001201" @rlad<"20001201"	US-PGPUB; USPAT	OR	OFF	2005/08/09 15:59
S28	50	S27 and S26	US-PGPUB; USPAT	OR	OFF	2005/08/09 15:59
S29	18	("5193071" \| "5276681" \| "5544329" \| "5918073" \| "5918248" \| "5982776" \| "5983311" \| "5999531" \| "6108336" \| "6125413" \| "6134607" \| "6160812" \| "6286076" \| "6295575" \| "6360287" \| "6393548" \| "6463032" \| "6509988").PN.	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/09 16:00
S30	250	load adj balanc$3 and S21 and S27	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/09 16:00
S31	121	S24 and S30	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:04
S32	62	BESHAI.inv.	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:04
S33	57430	(multiple near2 channel$1) multi$1channel$1	US-PGPUB; USPAT	OR	OFF	2005/08/10 13:04

S34	25	S33 and S32	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:16
S35	0	S32 and delsy.clm.	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:17
S36	24	S32 and delay.clm.	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:27
S37	14	S36 not S34	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:27
S38	2916	nortel.as.	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:27
S39	215	S38 and delay.clm.	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:27
S40	24	S33 and S39	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:27
S41	19	S40 not S32	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:29
S42	5047667	(sum$4 summation add$3 addition)	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:30
S43	691593	latenc$3 propagation$1 delay$1 bandiwdth$1	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:31
S44	634000	S42 and S43	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:31
S45	193538	S42 same S43	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:33
S46	3407594	@ad<"20001201" @rlad<"20001201"	US-PGPUB; USPAT	OR	OFF	2005/08/10 13:33
S47	106773	S45 and S46	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:33
S48	3407594	S46 and S46	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:33
S49	6451	S47 and S33	US-PGPUB; USPAT; USOCR	OR	OFF	2005/08/10 13:33

S50	844	composite near2 index	US-PGPUB; USPAT	OR	OFF	2005/08/10 13:34
S51	91	merit near2 index	US-PGPUB; USPAT	OR	OFF	2005/08/10 13:34
S52	3	S50 and S51	US-PGPUB; USPAT	OR	OFF	2005/08/10 13:39
S53	0	load adj balanc$3 and delay.clm. and propagation.clm. and latenc$3.clm.	US-PGPUB; USPAT	OR	OFF	2005/08/10 13:39
S54	57	delay.clm. and propagation.clm. and latenc$3.clm.	US-PGPUB; USPAT	OR	OFF	2005/08/10 13:41
S55	0	S32 and S54	US-PGPUB; USPAT	OR	OFF	2005/08/10 13:40
S56	0	S38 and S54	US-PGPUB; USPAT	OR	OFF	2005/08/10 13:41
S57	57	delay.clm. and propagation.clm. and latenc$3.clm.	US-PGPUB; USPAT	OR	OFF	2005/08/10 13:41
S58	29	delay.clm. and propagation.clm. and latenc$3.clm.	US-PGPUB	OR	OFF	2005/08/10 14:29

PORTAL USPTO

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: The ACM Digital Library The Guide

+vacancy +latency +delay +load +balancing

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used **vacancy latency delay load balancing** Found **2** of **158,639**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 2 of 2

Relevance scale

1 Prefetching using Markov predictors

Doug Joseph, Dirk Grunwald

May 1997 **ACM SIGARCH Computer Architecture News , Proceedings of the 24th annual international symposium on Computer architecture**, Volume 25 Issue 2

Full text available: pdf(1.68 MB) Additional Information: full citation, abstract, references, citings, index terms

Prefetching is one approach to reducing the latency of memory operations in modern computer systems. In this paper, we describe the *Markov prefetcher.* This prefetcher acts as an interface between the on-chip and off-chip cache, and can be added to existing computer designs. The Markov prefetcher is distinguished by prefetching *multiple reference predictions* from the memory subsystem, and then prioritizing the delivery of those references to the processor.This design results in a pr ...

2 Using thread-level speculation to simplify manual parallelization

Manohar K. Prabhu, Kunle Olukotun

June 2003 **ACM SIGPLAN Notices , Proceedings of the ninth ACM SIGPLAN symposium on Principles and practice of parallel programming**, Volume 38 Issue 10

Full text available: pdf(440.50 KB) Additional Information: full citation, abstract, references, citings, index terms

In this paper, we provide examples of how thread-level speculation (TLS) simplifies manual parallelization and enhances its performance. A number of techniques for manual parallelization using TLS are presented and results are provided that indicate the performance contribution of each technique on seven SPEC CPU2000 benchmark applications. We also provide indications of the programming effort required to parallelize each benchmark. TLS parallelization yielded a 110% speedup on our four ...

Keywords: chip multiprocessor, data speculation, feedback-driven optimization, manual parallel programming, multithreading

Results 1 - 2 of 2

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.

Terms of Usage Privacy Policy Code of Ethics Contact Us

Useful downloads: Adobe Acrobat QuickTime Windows Media Player Real Player

PORTAL USPTO

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: The ACM Digital Library The Guide

+bandwidth +latency +delay +load +balance

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used **bandwidth latency delay load balance** Found **1,306** of **158,639**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown Relevance scale

1 Efficient load balancing for wide-area divide-and-conquer applications

Rob V. van Nieuwpoort, Thilo Kielmann, Henri E. Bal

June 2001 **ACM SIGPLAN Notices , Proceedings of the eighth ACM SIGPLAN symposium on Principles and practices of parallel programming**, Volume 36 Issue 7

Full text available: pdf(118.89 KB) Additional Information: full citation, abstract, references, citings, index terms

Divide-and-conquer programs are easily parallelized by letting the programmer annotate potential parallelism in the form of spawn and sync constructs. To achieve efficient program execution, the generated work load has to be balanced evenly among the available CPUs. For single cluster systems, Random Stealing (RS) is known to achieve optimal load balancing. However, RS is inefficient when applied to hierarchical wide-area systems where m ...

Keywords: Java, clustered wide-area systems, distributed supercomputing

2 A new method to make communication latency uniform: distributed routing balancing

D. Franco, I. Garcés, E. Luque

May 1999 **Proceedings of the 13th international conference on Supercomputing**

Full text available: pdf(1.24 MB) Additional Information: full citation, references, citings, index terms

Keywords: adaptive routing, distributed routing balancing, hot spot avoidance, interconnection networks, random routing, traffic distribution, uniform latency

3 Processor microarchitecture I: Partitioned first-level cache design for clustered microarchitectures

Paul Racunas, Yale N. Patt

June 2003 **Proceedings of the 17th annual international conference on Supercomputing**

Full text available: pdf(191.74 KB) Additional Information: full citation, abstract, references, citings, index terms

The high clock frequencies of modern superscalar processors make the wire delay incurred in moving data across the processor chip a significant concern. As frequencies continue to increase, it will become more difficult for a centralized first level data cache to supply the

timely data bandwidth required by superscalar processors.This paper presents a complete solution for the partitioning of the first level of the memory hierarchy. The first level data cache is split into several independent pa ...

Keywords: clustered microarchitecture, partitioned cache

4 Improved methods for hiding latency in high bandwidth networks (extended abstract)
Matthew Andrews, Tom Leighton, P. Takis Metaxas, Lisa Zhang
June 1996 **Proceedings of the eighth annual ACM symposium on Parallel algorithms and architectures**
Full text available: pdf(1.11 MB) Additional Information: full citation, references, citings, index terms

5 Accounting for memory bank contention and delay in high-bandwidth multiprocessors
Guy E. Blelloch, Phillip B. Gibbons, Yossi Matias, Marco Zagha
July 1995 **Proceedings of the seventh annual ACM symposium on Parallel algorithms and architectures**
Full text available: pdf(1.26 MB) Additional Information: full citation, references, citings, index terms

6 Traffic-based Load Balance for Scalable Network Emulation
Xin Liu, Andrew A. Chien
November 2003 **Proceedings of the 2003 ACM/IEEE conference on Supercomputing**
Full text available: pdf(191.58 KB) Additional Information: full citation, abstract

Load balance is critical to achieving scalability for large network emulation studies, which are of compelling interest for emerging Grid, Peer to Peer, and other distributed applications and middleware. Achieving load balance in emulation is difficult because of irregular network structure and unpredictable network traffic. We formulate load balance as a graph partitioning problem and apply classical graph partitioning algorithms to it. The primary challenge in this approach is how to extract u ...

7 Simulation of a DQDB MAC protocol with movable boundary and bandwidth balancing mechanisms
S. Popovich, M. Alam, S. Bandyopadhyay
April 1992 **Proceedings of the 25th annual symposium on Simulation**
Full text available: pdf(1.00 MB) Additional Information: full citation, references, index terms

8 GOAL: a load-balanced adaptive routing algorithm for torus networks
Arjun Singh, William J. Dally, Amit K. Gupta, Brian Towles
May 2003 **ACM SIGARCH Computer Architecture News , Proceedings of the 30th annual international symposium on Computer architecture**, Volume 31 Issue 2
Full text available: pdf(260.66 KB) Additional Information: full citation, abstract, references, citings

We introduce a load-balanced adaptive routing algorithm for torus networks, GOAL - Globally Oblivious Adaptive Locally - that provides high throughput on adversarial traffic patterns, matching or exceeding fully randomized routing and exceeding the worst-case performance of Chaos [2], RLB [14], and minimal routing [8] by more than 40%. GOAL also preserves locality to provide up to 4.6× the throughput of fully randomized routing [19] on local traffic. GOAL achieves global load balance by ra ...

9 Balanced scheduling: instruction scheduling when memory latency is uncertain
Daniel R. Kerns, Susan J. Eggers
June 1993 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1993 conference on Programming language design and implementation**, Volume 28 Issue 6
Full text available: pdf(1.24 MB) Additional Information: full citation, abstract, references, citings, index terms

Traditional list schedulers order instructions based on an optimistic estimate of the load delay imposed by the implementation. Therefore they cannot respond to variations in load latencies (due to cache hits or misses, congestion in the memory interconnect, etc.) and cannot easily be applied across different implementations. We have developed an alternative algorithm, known as balanced scheduling, that schedule instructions based on an estimate of the amount of instruction level parallelis ...

10 Comparing random data allocation and data striping in multimedia servers
Jose Renato Santos, Richard R. Muntz, Berthier Ribeiro-Neto
June 2000 **ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 2000 ACM SIGMETRICS international conference on Measurement and modeling of computer systems**, Volume 28 Issue 1
Full text available: pdf(1.18 MB) Additional Information: full citation, abstract, references, citings, index terms

We compare performance of a multimedia storage server based on a random data allocation layout and block replication with traditional data striping techniques. Data striping techniques in multimedia servers are often designed for restricted workloads, e.g. sequential access patterns with CBR (constant bit rate) requirements. On the other hand, a system based on random data allocation can support virtually any type of multimedia application, including VBR (variable bit rate) video or audio, ...

11 An adaptive, non-uniform cache structure for wire-delay dominated on-chip caches
Changkyu Kim, Doug Burger, Stephen W. Keckler
October 2002 **Proceedings of the 10th international conference on Architectural support for programming languages and operating systems**, Volume 37 , 30 , 36 Issue 10 , 5 , 5
Full text available: pdf(1.33 MB) Additional Information: full citation, abstract, references, citings

Growing wire delays will force substantive changes in the designs of large caches. Traditional cache architectures assume that each level in the cache hierarchy has a single, uniform access time. Increases in on-chip communication delays will make the hit time of large on-chip caches a function of a line's physical location within the cache. Consequently, cache access times will become a continuum of latencies rather than a single discrete latency. This non-uniformity can be exploited to provide ...

12 Rotating combined queueing (RCQ): bandwidth and latency guarantees in low-cost, high-performance networks
Jae H. Kim, Andrew A. Chien
May 1996 **ACM SIGARCH Computer Architecture News , Proceedings of the 23rd annual international symposium on Computer architecture**, Volume 24 Issue 2
Full text available: pdf(1.39 MB) Additional Information: full citation, abstract, references, citings, index terms

Network service guarantees not only provide significant performance benefits to distributed computing systems (more balanced resource utilization, fast fault recovery, and fair network access), but they are also essential for many new applications requiring real-time communications with continuous data types (audio/video). Most existing algorithms which provide network service guarantees are too complicated to be feasible in high-speed, low-cost switches for multicomputer networks. The simpler a ...

13 Load-balanced location management for cellular mobile systems using quorums and dynamic hashing

Ravi Prakash, Zygmunt Haas, Mukesh Singhal

September 2001 **Wireless Networks**, Volume 7 Issue 5

Full text available: pdf(282.87 KB) Additional Information: full citation, abstract, references, index terms

This paper presents a new distributed location management strategy for cellular mobile systems. Its salient features are fast location update and query, load balancing among location servers, and scalability. The strategy employs dynamic hashing techniques and quorums to manage location update and query operations. The proposed strategy does not require a home location register (HLR) to be associated with each mobile node. Location updates and queries for a mobile node are multicast to subsets o ...

Keywords: distributed location management, dynamic hashing, location independent numbering, mobile computing, quorum systems, tries

14 Effects of communication latency, overhead, and bandwidth in a cluster architecture

Richard P. Martin, Amin M. Vahdat, David E. Culler, Thomas E. Anderson

May 1997 **ACM SIGARCH Computer Architecture News , Proceedings of the 24th annual international symposium on Computer architecture**, Volume 25 Issue 2

Full text available: pdf(2.08 MB) Additional Information: full citation, abstract, references, citings, index terms

This work provides a systematic study of the impact of communication performance on parallel applications in a high performance network of workstations. We develop an experimental system in which the communication latency, overhead, and bandwidth can be independently varied to observe the effects on a wide range of applications. Our results indicate that current efforts to improve cluster communication performance to that of tightly integrated parallel machines results in significantly improved ...

15 Parallel and distributed systems and networking: Analysis of Distributed Routing Balancing behavior

I. Garcés, D. Franco

March 2002 **Proceedings of the 2002 ACM symposium on Applied computing**

Full text available: pdf(613.16 KB) Additional Information: full citation, abstract, references, index terms

Distributed Routing Balancing (DRB) is a method developed to uniformly balance communication traffic over the interconnection network. DRB takes a similar approach to communications as load balancing does to processes in a distributed environment. The key ideas behind DRB are to distribute communication load based on limited and load-controlled path expansion, in order to maintain low message latency. In this paper, we present an exhaustive evaluation of DRB that shows that the method presents l ...

Keywords: adaptive routing, distributed routing balancing, hot spot avoidance, interconnection networks, traffic distribution

16 Reducing wire delay penalty through value prediction

Joan-Manuel Parcerisa, Antonio González

December 2000 **Proceedings of the 33rd annual ACM/IEEE international symposium on Microarchitecture**

Full text available: pdf(148.85 KB) ps(401.62 KB) Publisher Site Additional Information: full citation, references, citings, index terms

17 Diffracting trees

Nir Shavit, Asaph Zemach

November 1996 **ACM Transactions on Computer Systems (TOCS)**, Volume 14 Issue 4

Full text available: pdf(729.57 KB) Additional Information: full citation, abstract, references, citings, index terms

Shared counters are among the most basic coordination structures in multiprocessor conputation, with applications ranging from barrier synchronization to concurrent-data-structure design. This article introduces diffracting trees, novel data structures for share counting and load balancing in a distributed/parallel environment. Empirical evidence, collected on a simulated distributed shared-memory machine and several simulated message-passing architectures, shows that diffracting trees scal ...

Keywords: contention, counting networks, index distribution, lock free, wait free

18 Fingerprinting: bounding soft-error detection latency and bandwidth

Jared C. Smolens, Brian T. Gold, Jangwoo Kim, Babak Falsafi, James C. Hoe, Andreas G. Nowatzyk

October 2004 **Proceedings of the 11th international conference on Architectural support for programming languages and operating systems**, Volume 39 , 32 , 38 Issue 11 , 5 , 5

Full text available: pdf(229.65 KB) Additional Information: full citation, abstract, references, index terms

Recent studies have suggested that the soft-error rate in microprocessor logic will become a reliability concern by 2010. This paper proposes an efficient error detection technique, called *fingerprinting*, that detects differences in execution across a dual modular redundant (DMR) processor pair. Fingerprinting summarizes a processor's execution history in a hash-based signature; differences between two mirrored processors are exposed by comparing their fingerprints. Fingerprinting tightly ...

Keywords: backwards error recovery (BER), dual modular redundancy (DMR), error detection, soft errors

19 Efficient fair queueing using deficit round-robin

M. Shreedhar, George Varghese

June 1996 **IEEE/ACM Transactions on Networking (TON)**, Volume 4 Issue 3

Full text available: pdf(1.57 MB) Additional Information: full citation, references, citings, index terms

20 A two-tier heterogeneous mobile Ad Hoc network architecture and its load-balance routing problem

Chi-Fu Huang, Hung-Wei Lee, Yu-Chee Tseng

August 2004 **Mobile Networks and Applications**, Volume 9 Issue 4

Full text available: pdf(1.28 MB) Additional Information: full citation, abstract, references, index terms

The mobile ad hoc network (MANET) has attracted a lot of interest recently. However, most of the existing works have assumed a stand-alone MANET. In this paper, we propose a two-tier, heterogeneous MANET architecture which can support Internet access. The low tier of the network consists of a set of mobile hosts each equipped with a IEEE 802.11 wireless LAN card. In order to connect to the Internet and handle the network partitioning problem, we propose that the high tier is comprised of a subse ...

Keywords: ad hoc network, load balance, mobile computing, routing, wireless network

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.
Terms of Usage Privacy Policy Code of Ethics Contact Us

Useful downloads: Adobe Acrobat QuickTime Windows Media Player Real Player